95-702 Distributed Systems

Networking

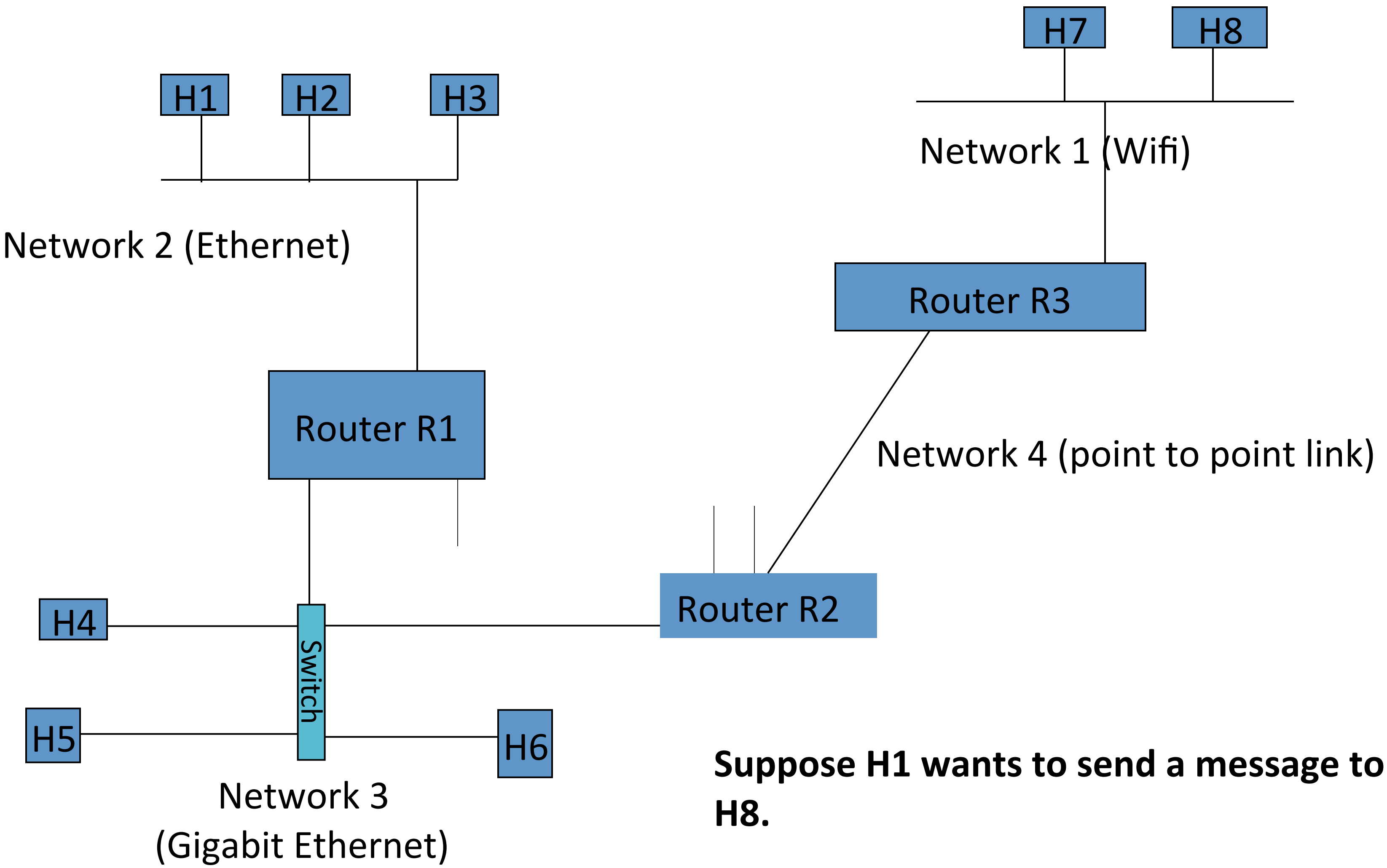
Learning Goals

- Be comfortable with terminology used concerning the Internet
- Understand the role of protocols, and the layering of protocols, in the architecture of the Internet. And how this layering provides levels of abstraction below which a developer need not be (too) concerned.
- Understand the basic functionality of how packets of information travel between one system and another. This will inform design and configuration choices in building and maintaining systems.
- Understand IP addressing.

Coulouris Text

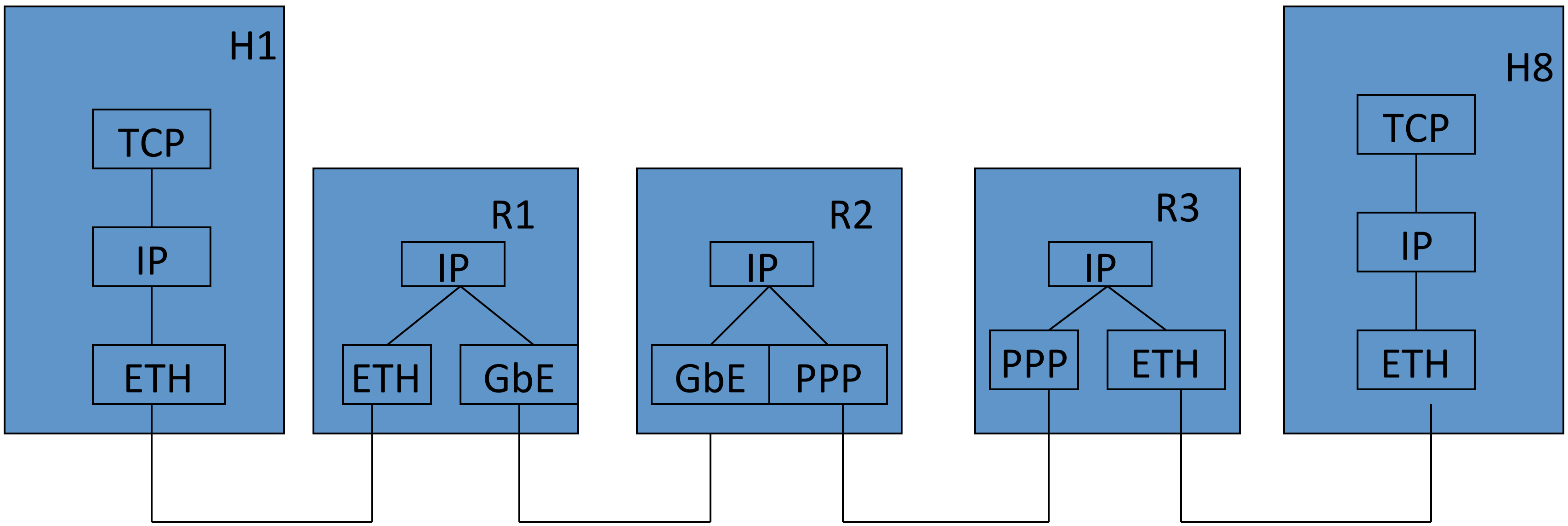
- For the exams, be sure you have read Ch 3 – Networking and Internetworking through pg 128.
- We will not test on the case studies pg 128+, but they are worthwhile reading.
- Be sure to understand all terminology.
- I will add a test quiz to Blackboard.
  - It is **not** all inclusive of the questions we will ask.
  - But it does give you an idea of the type of questions.

A Simple Example Internetwork



Suppose H1 wants to send a message to H8.

H1 To H8



Protocol Layering

We will start at the lowest layer and move up

MAC address

- Each networking device has a MAC address
  - Media Access Control address
  - This has nothing to do with Apple, Inc.
- Fixed by manufacturer
  - (Can be spoofed, for good or by Eve/Mallory)
- My laptop has 3:
  - Ethernet: 60:fb:42:ff:fe:f8:b5:08
  - Wifi: 60:fb:42:f8:b5:08
  - Bluetooth: 60:fb:42:72:08:4c
- My iPhone has 2:
  - Wifi
  - Bluetooth
  - And an IMEI number, which is like a mobile MAC

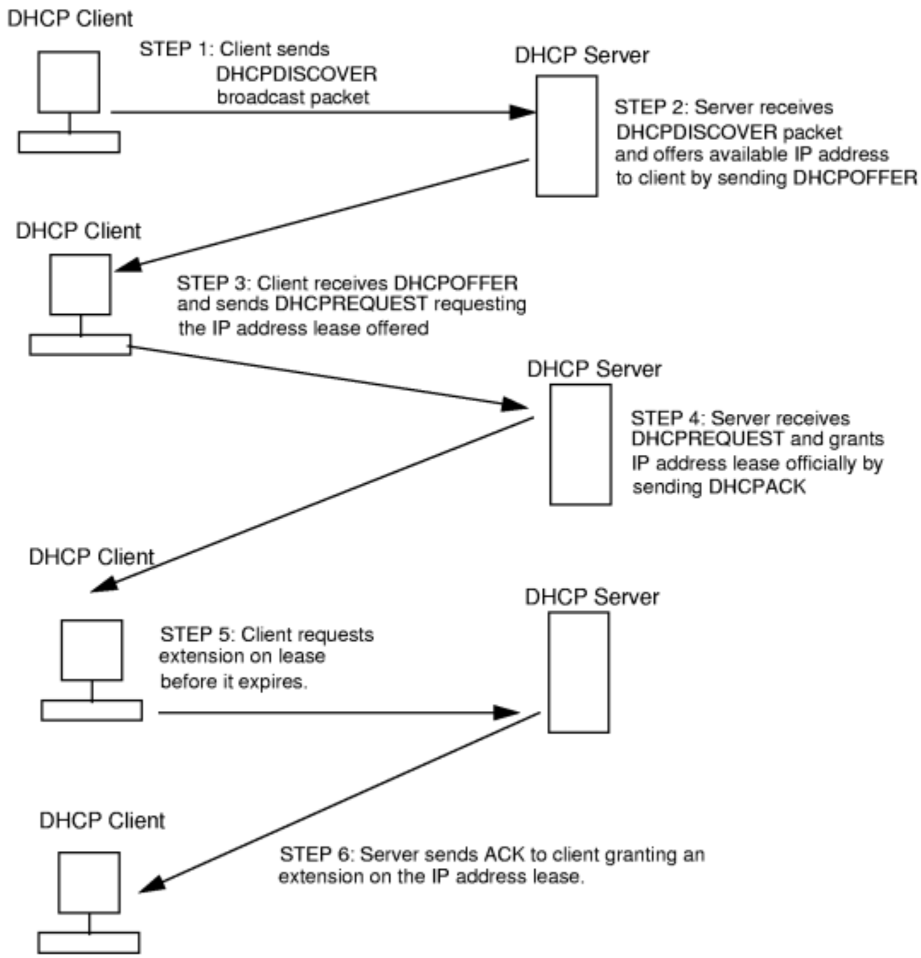
Important distinction

- MAC addresses are used to pass messages around a single physical network segment
  - An ethernet segment is a network
  - A wifi access point forms a wireless network segment
  - A point-to-point microwave link is a network segment
- IP addresses are used to pass messages around between networks
  - I.e. between the wifi LAN in this room and google.com
  - IP == Internet Protocol
    - Internet meaning between networks

Where do IP addresses come from?

- So your MAC address is in hardware
- Where does your IP address come from?
  - Static – from a network administrator
  - Dynamic – Dynamic Host Configuration Protocol

DHCP Negotiation



To broadcast,
send to IP address:
255.255.255.255
which datalink sends
to MAC address:
FF:FF:FF:FF:FF:FF

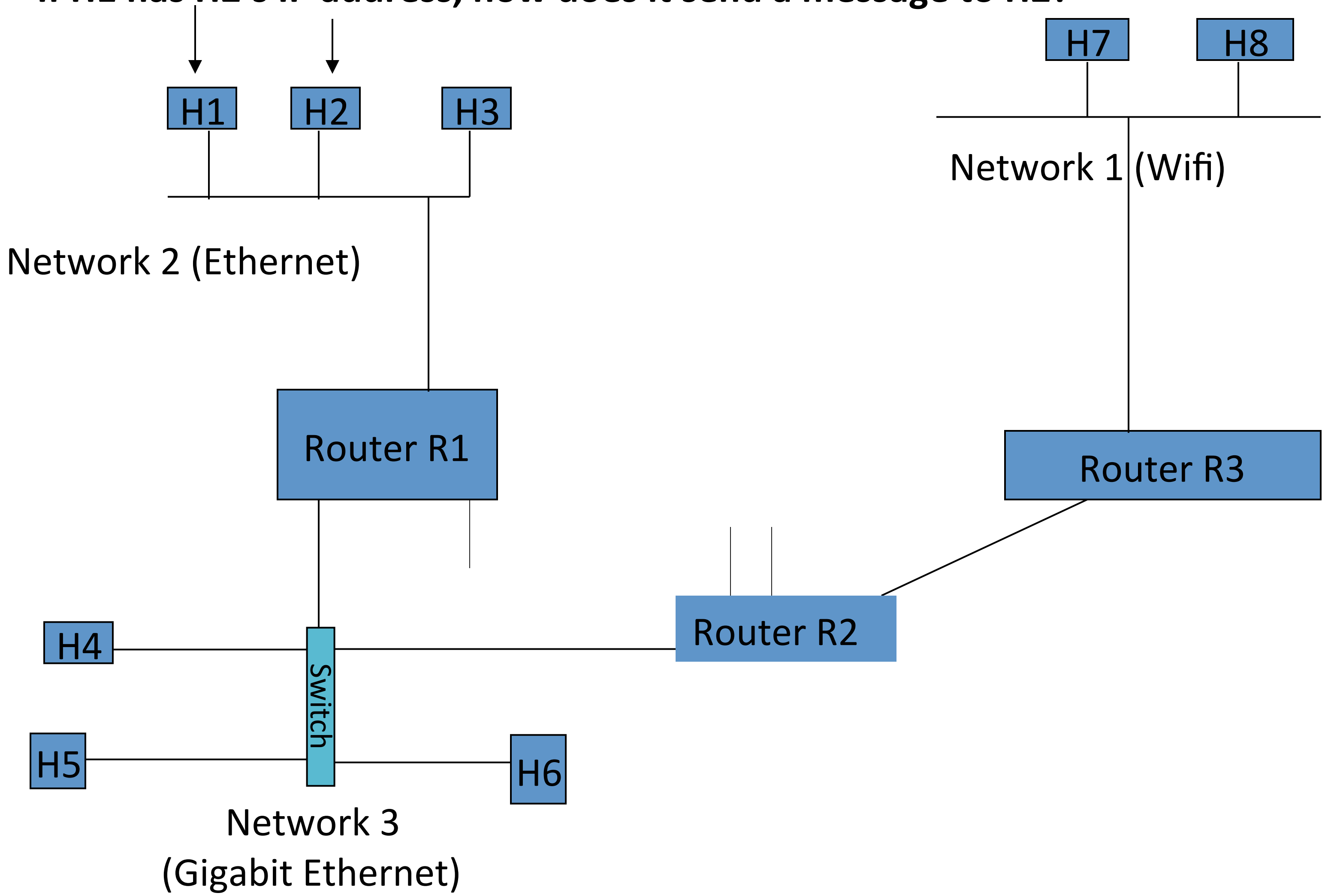
Source: http://docs.hp.com/en/B2355-90685/ch06s02.html

DHCP Offer

- IP Address being offerred to the client
- Subnet mask for the local network
- Gateway IP (default router out of this network)
- DNS servers
- Lease duration

Network addressing – within a network

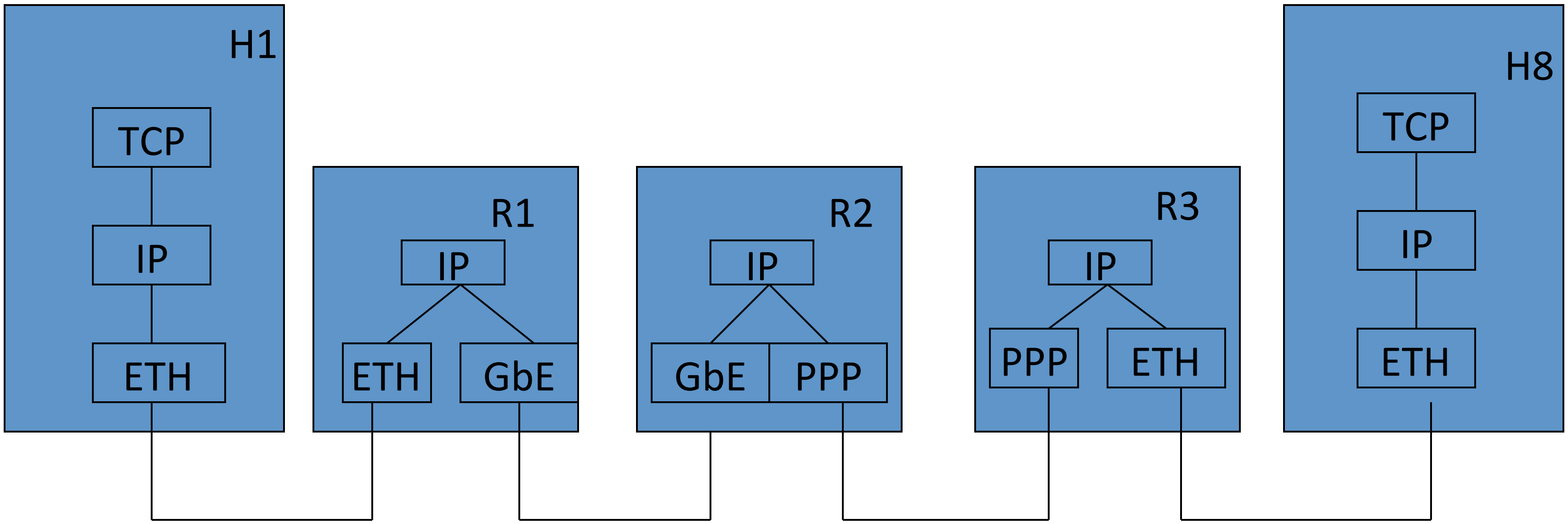
IP addresses are needed for application addressing of even local hosts (e.g. office printer)
If H1 has H2's IP address, how does it send a message to H2?



Address Resolution Protocol (ARP)

- Host H1 wants to contact host H2 on the same network.
- First, H1 checks its cache to see if it already contains the IP / MAC address pair.
  - If it does then use the MAC address.
- If it does not
  - H1 broadcasts the IP address to all hosts on this network.
    - Broadcast MAC is FF:FF:FF:FF:FF:FF
  - The matching host H2 sends back its MAC address.
  - H1 then adds this mapping to its cache.
- Try in command-line window *arp -a*

H1 To H8



Now lets move up a layer
To the Internet Protocol (IP) layer

Inter-network addressing

- Assume you are an http request from your laptop to a server at 202.65.43.185
- Surmise:
  - How can you get from your browser to this server.

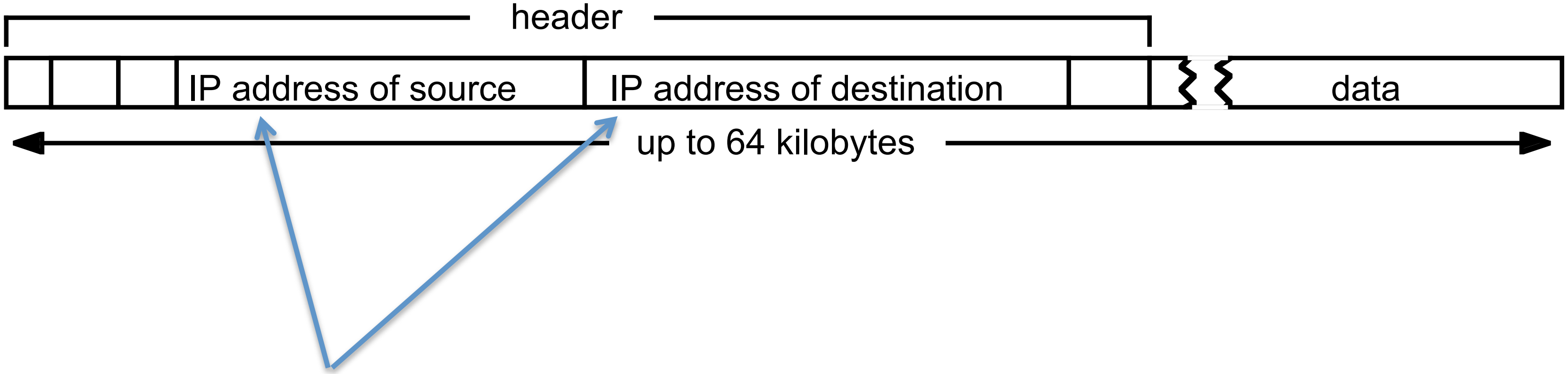
Route to 202.65.43.185

```
nut:~ JoeMertz$ traceroute -I health.gov.ck
traceroute to health.gov.ck (202.65.43.185), 64 hops max, 72 byte packets
 1  pod-a-weh-vl88.gw.cmu.net (128.2.80.3)  1.094 ms  0.652 ms  0.637 ms
 2  core255-vl919.gw.cmu.net (128.2.255.161)  0.934 ms  0.688 ms  0.677 ms
 3  pod-i-nh-vl987.gw.cmu.net (128.2.255.251)  0.929 ms  0.873 ms  0.823 ms
 4  ge-7-23.car1.pittsburgh3.level3.net (4.49.108.45)  1.174 ms  1.169 ms  1.114 ms
 5  ae-5-5.ebr1.washington1.level3.net (4.69.135.242)  17.289 ms  17.162 ms  17.947 ms
 6  ae-91-91.csw4.washington1.level3.net (4.69.134.142)  19.895 ms  16.998 ms  18.090 ms
 7  ae-4-99.edge2.washington4.level3.net (4.68.17.211)  7.646 ms  7.966 ms  7.682 ms
 8  if-8-0.icore1.aeq-ashburn.as6453.net (206.82.139.65)  12.407 ms  16.747 ms  18.806 ms
 9  if-0-0-0-459.core4.aeq-ashburn.as6453.net (216.6.42.21)  7.971 ms  8.485 ms  8.276 ms
10  if-3-0-0-921.core1.ct8-chicago.as6453.net (66.110.27.53)  20.337 ms  22.613 ms  20.412 ms
11  if-0-0-0-925.core1.s00-seattle.as6453.net (207.45.206.1)  91.743 ms  82.861 ms  81.951 ms
12  if-5-0.core1.vcw-vancouver.as6453.net (207.45.196.54)  83.803 ms  83.718 ms  83.970 ms
13  if-0-0.core1.lcn-lakecowichan.as6453.net (216.6.58.5)  86.043 ms  86.134 ms  86.489 ms
14  if-5-0-0.bb2.lcn-lakecowichan.as6453.net (64.86.83.39)  89.394 ms  88.493 ms  87.112 ms
15  ix-10-1-0.bb2.lcn-lakecowichan.as6453.net (64.86.84.2)  652.794 ms  605.698 ms  628.306 ms
16  202.65.32.18 (202.65.32.18)  624.766 ms  642.546 ms  621.597 ms
17  thealth.oyster.net.ck (202.65.43.185)  624.677 ms  740.438 ms  683.941 ms
```

Network vs Internetwork

- An internetwork is an interconnected collection of networks.
- The Internet Protocol (IP) is the key tool used today to build scalable, heterogeneous internetworks

IP packet layout

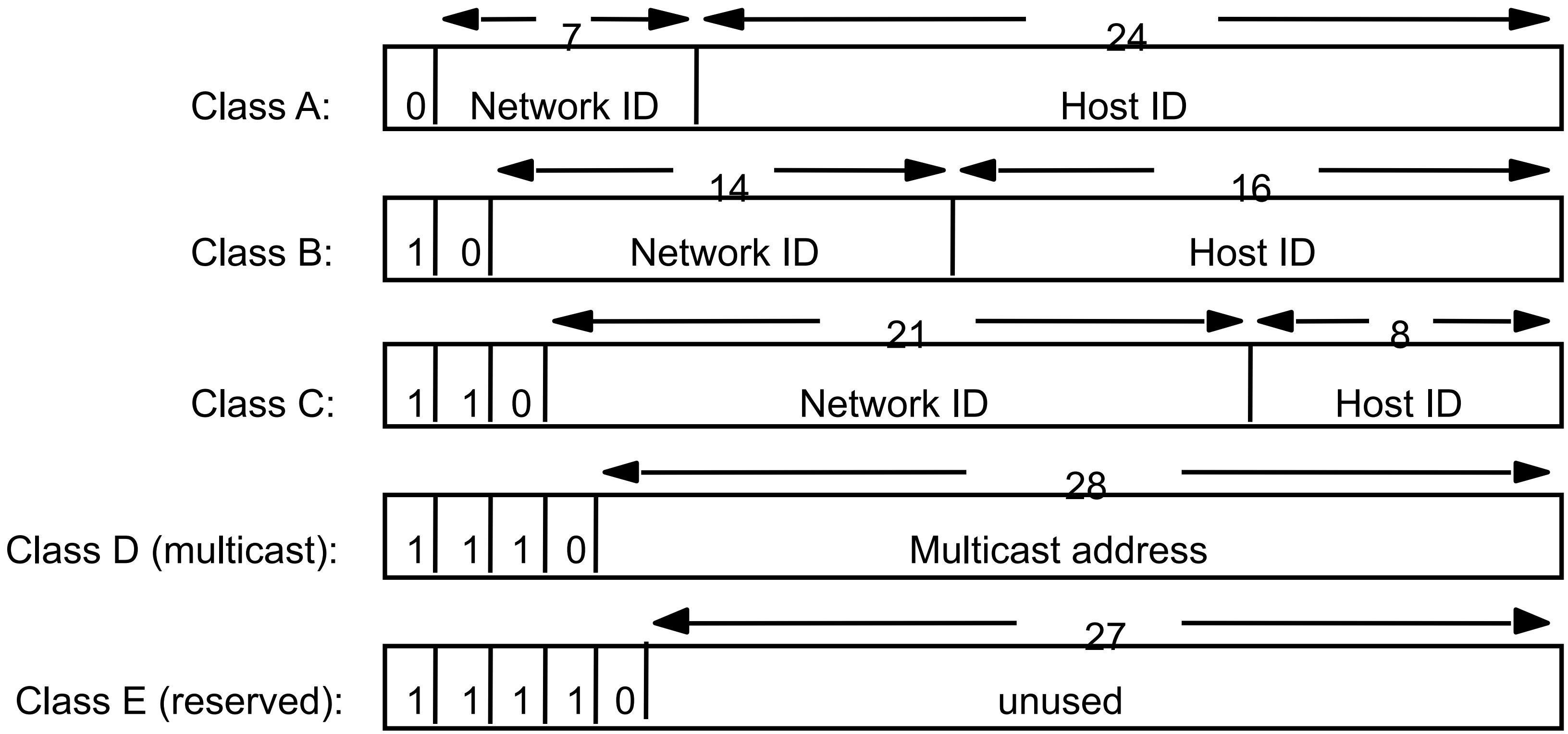


These must address every host on the planet!

(Sometimes we can cheat (see NAT) in Coulouris.)

32-bit IPv4 Addressing

IPv4 is the most common addressing today.



Decimal representation of IP addresses

E.g. IBM 9.0.0.0-9.255.255.255

E.g. CMU 128.2.0.0-128.2.255.255

	octet 1	octet 2	octet 3		Range of addresses
	Network ID	Host ID			
Class A:	1 to 127	0 to 255	0 to 255	0 to 255	1.0.0.0 to 127.255.255.255
	Network ID		Host ID		
Class B:	128 to 191	0 to 255	0 to 255	0 to 255	128.0.0.0 to 191.255.255.255
	Network ID			Host ID	
Class C:	192 to 223	0 to 255	0 to 255	1 to 254	192.0.0.0 to 223.255.255.255
	Multicast address				
Class D (multicast):	224 to 239	0 to 255	0 to 255	1 to 254	224.0.0.0 to 239.255.255.255
Class E (reserved):	240 to 255	0 to 255	0 to 255	1 to 254	240.0.0.0 to 255.255.255.255

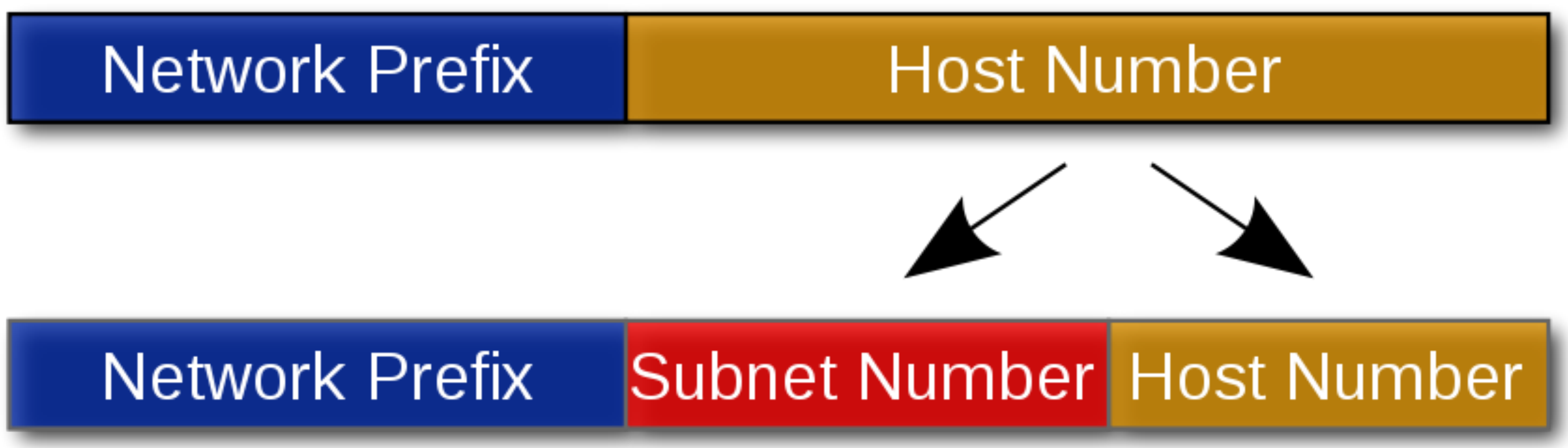
Test question: Who allocates IP addresses?

IP Addressing

- Every IP datagram contains the IP address of the destination host.
- The network prefix of an IP address uniquely identifies a ***single network*** that is part of the larger Internet.
- All hosts and routers that share the same network prefix of their address are connected to the same network and can thus communicate with each other by sending frames over the network, addressed with MAC addresses.
- The network prefix of the IP address, therefore, helps a packet find the right network.
- Every network that is part of the Internet has at least one router that, by definition, is also connected to at least one other network; this router can exchange packets with hosts or routers on either network.

Subnet Addressing

- Subnetting allows you to break your address space into smaller networks.



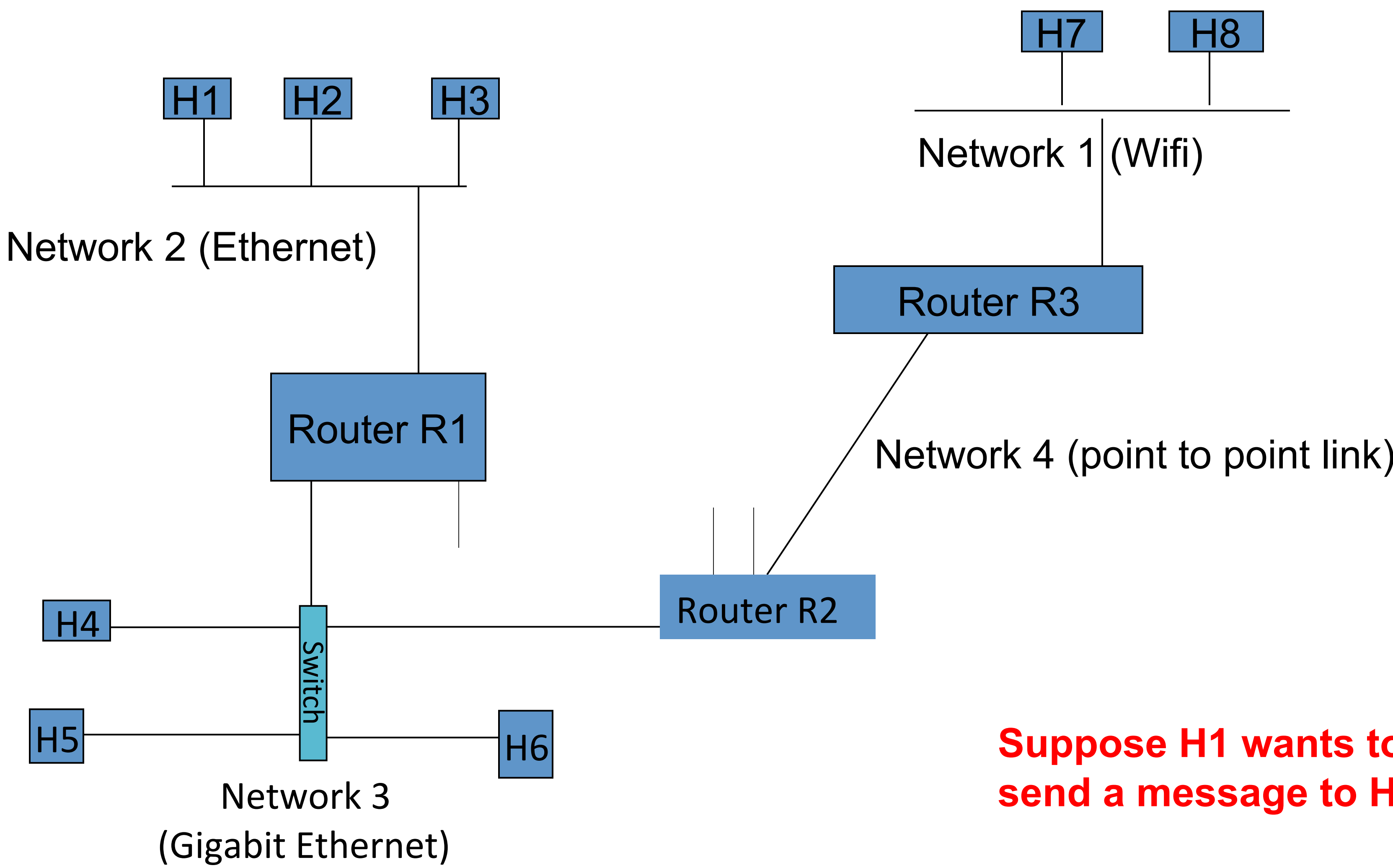
Source: http://en.wikipedia.org/wiki/Subnetwork

Subnet Addressing Example

	Dot-decimal notation	Binary form
IP address	192.168.5.130	11000000.10101000.00000101.10000010
Subnet Mask	255.255.255.0	11111111.11111111.11111111.00000000
Network Portion	192.168.5.0	11000000.10101000.00000101.00000000
Host Portion	0.0.0.130	00000000.00000000.00000000.10000010

- Similarly within CMU
  - Instead of one network 128.2 with 65K hosts
  - Network admins can use subnet masks to break it into multiple sub-networks (subnets).
    - A few top bits of the 3rd byte can be the subnet.
    - The remaining bits will be the host.
    - **Note that this will not always fall only on "dot" boundaries.**

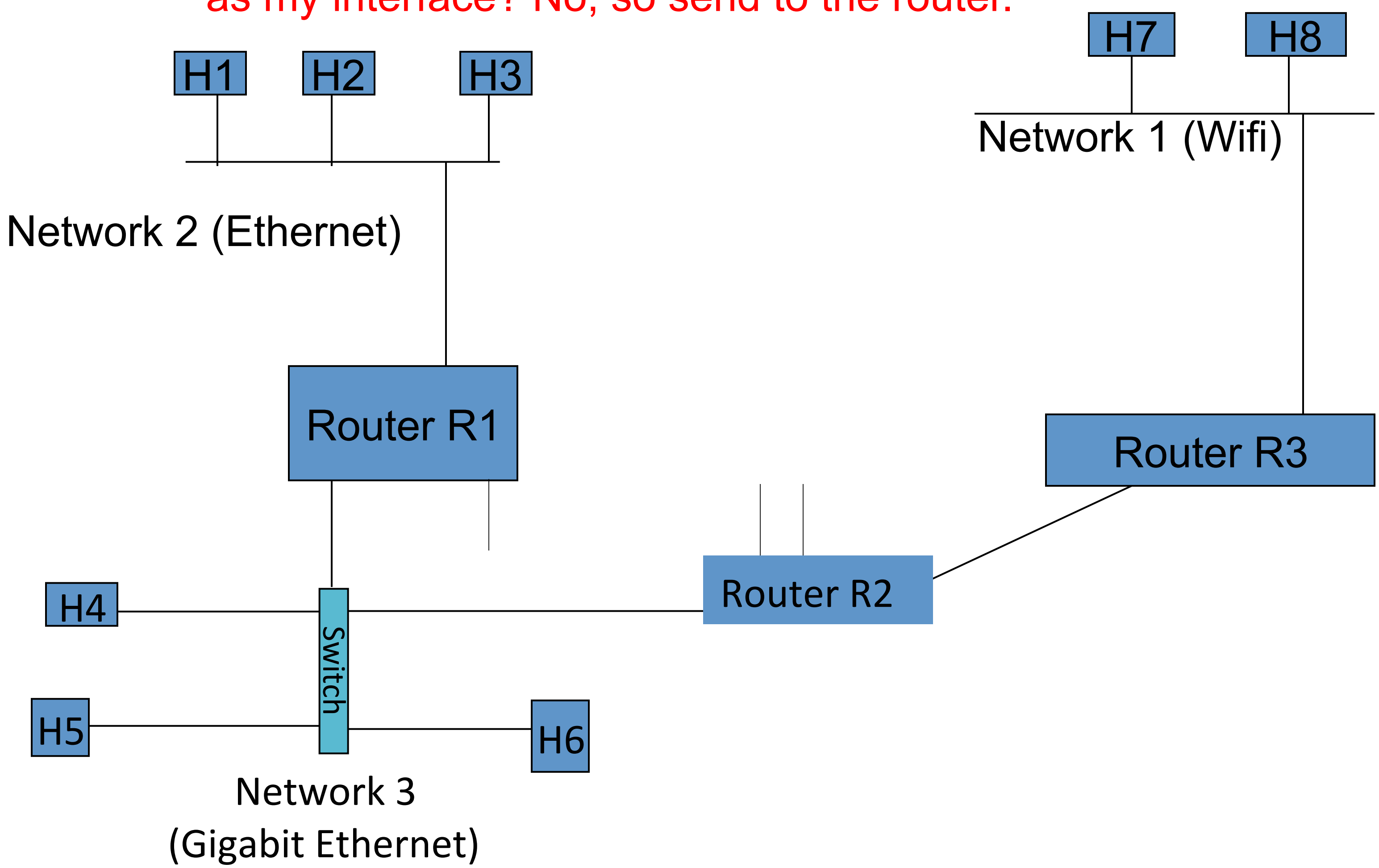
Example Internetwork



Suppose H1 wants to send a message to H8.

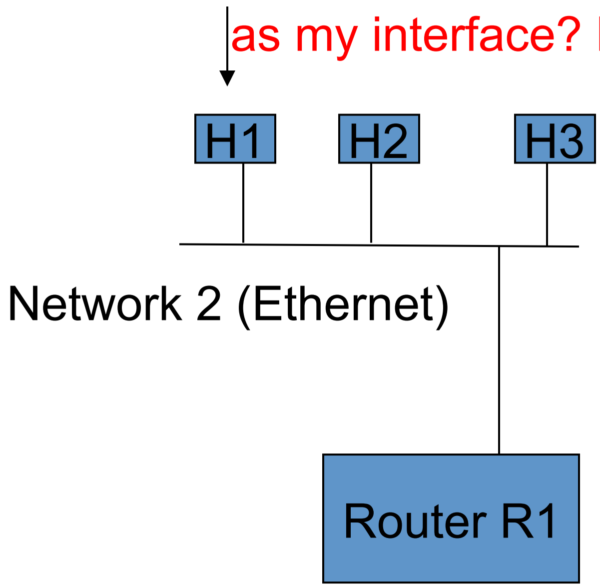
Choose R1

H1 has the IP address of H8. Does H8 have the same network prefix address as my interface? No, so send to the router.



Submasks

H1 has the IP address of H8. Does H8 have the same network prefix address as my interface? No, so send to the router.



But, how is this decision made?

Suppose Network 2 has a subnet mask of 255.255.255.0.

The leftmost 24 bits represent the network identifier. The remaining 8 bits represent the 2^8 (256) hosts.

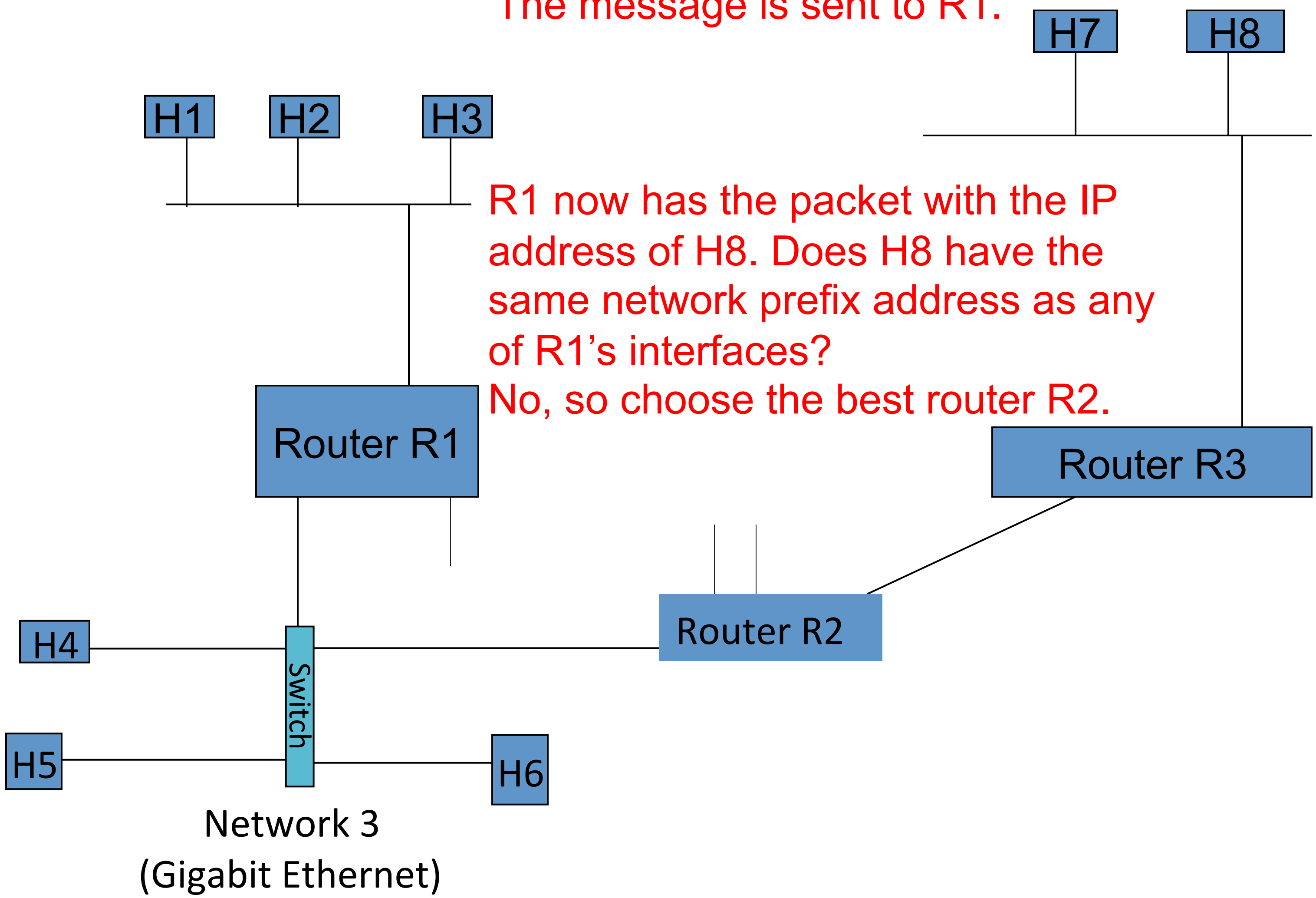
This is also known as a "/24 network"

H1 performs a bitwise AND of the subnet mask with H8's 32-bit IP address.

If the result does not match H1's network Identifier then H8 is a foreign machine.

R2

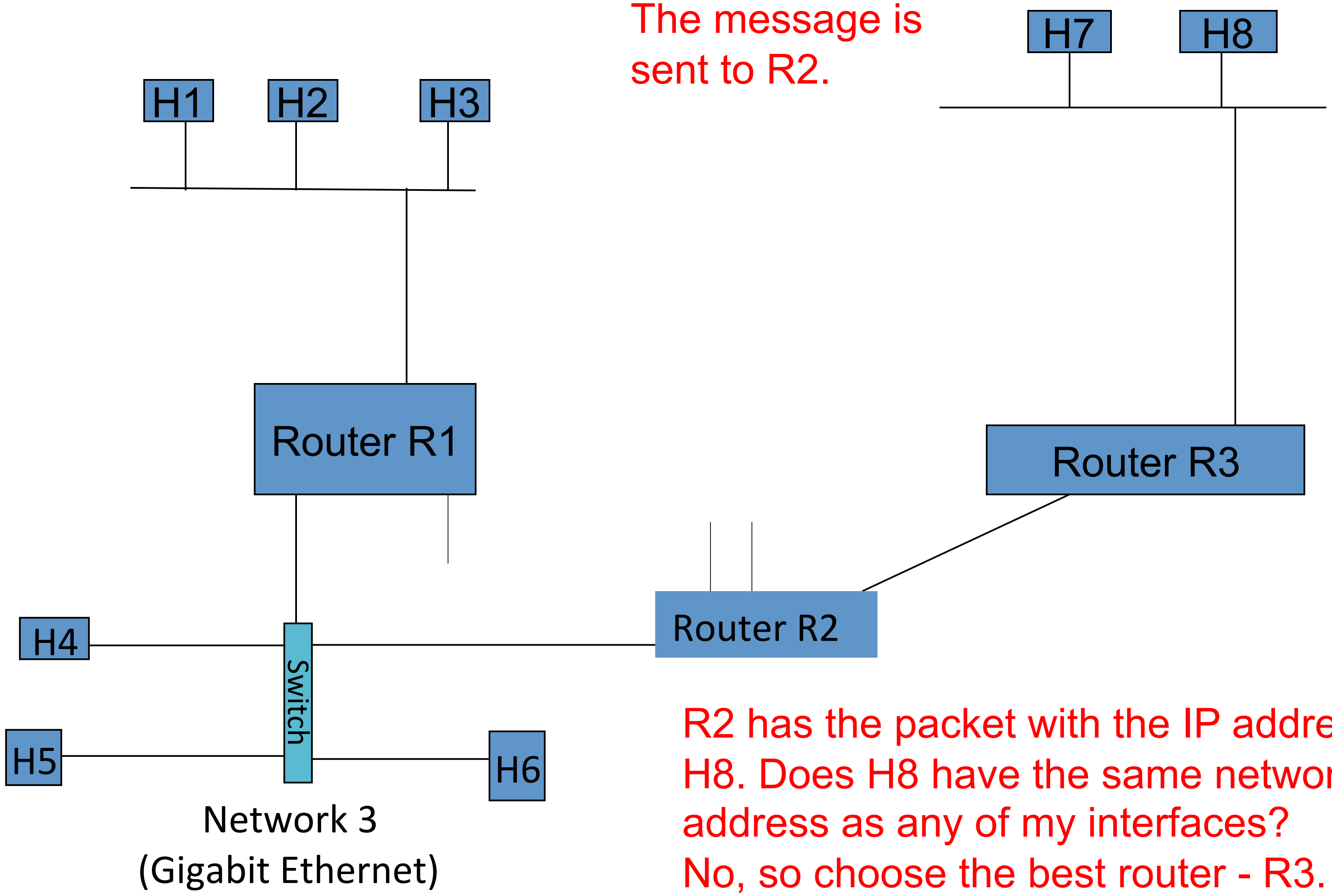
The message is sent to R1.



R1 now has the packet with the IP address of H8. Does H8 have the same network prefix address as any of R1's interfaces?
No, so choose the best router R2.

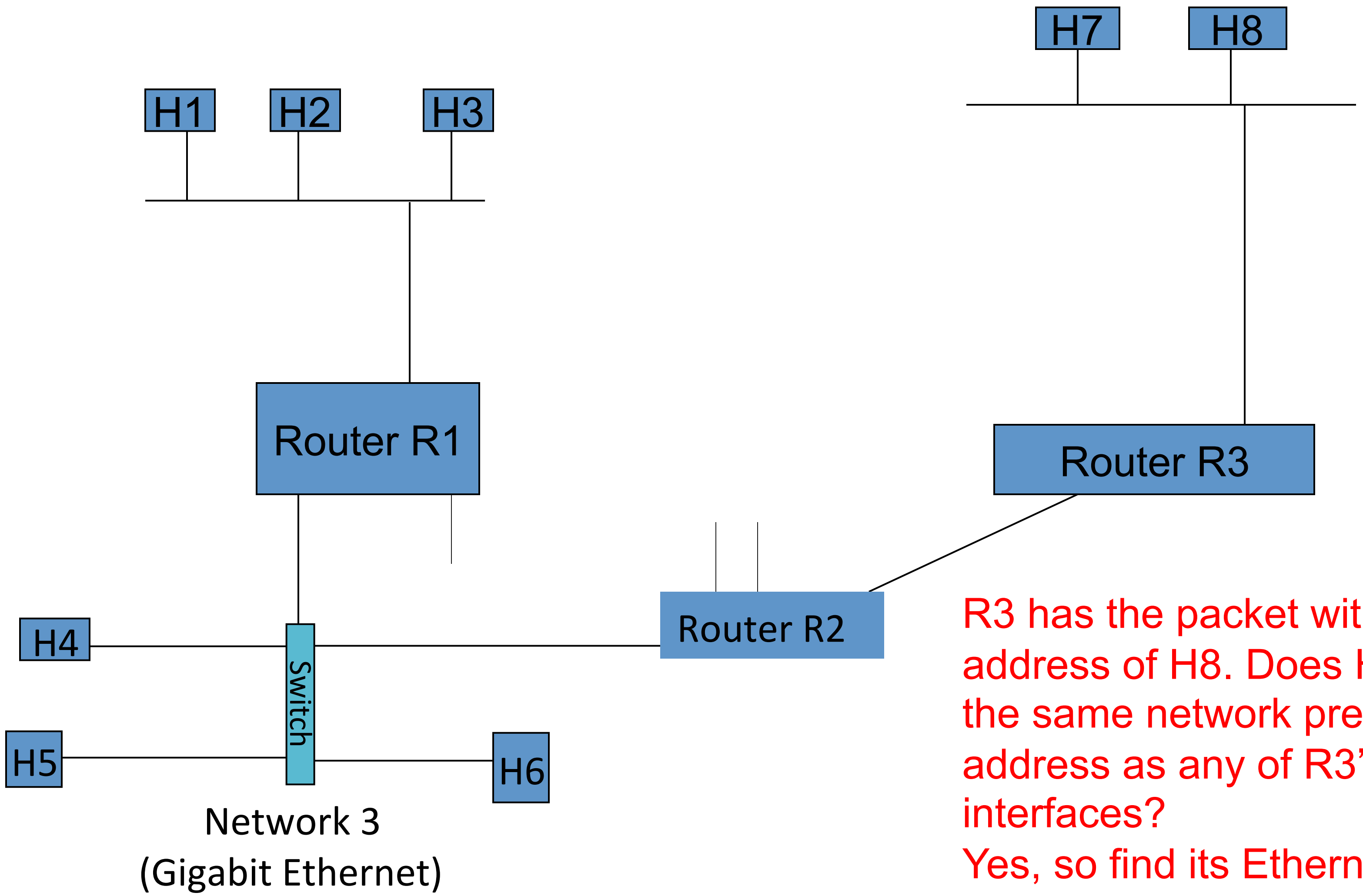
R3

The message is sent to R2.



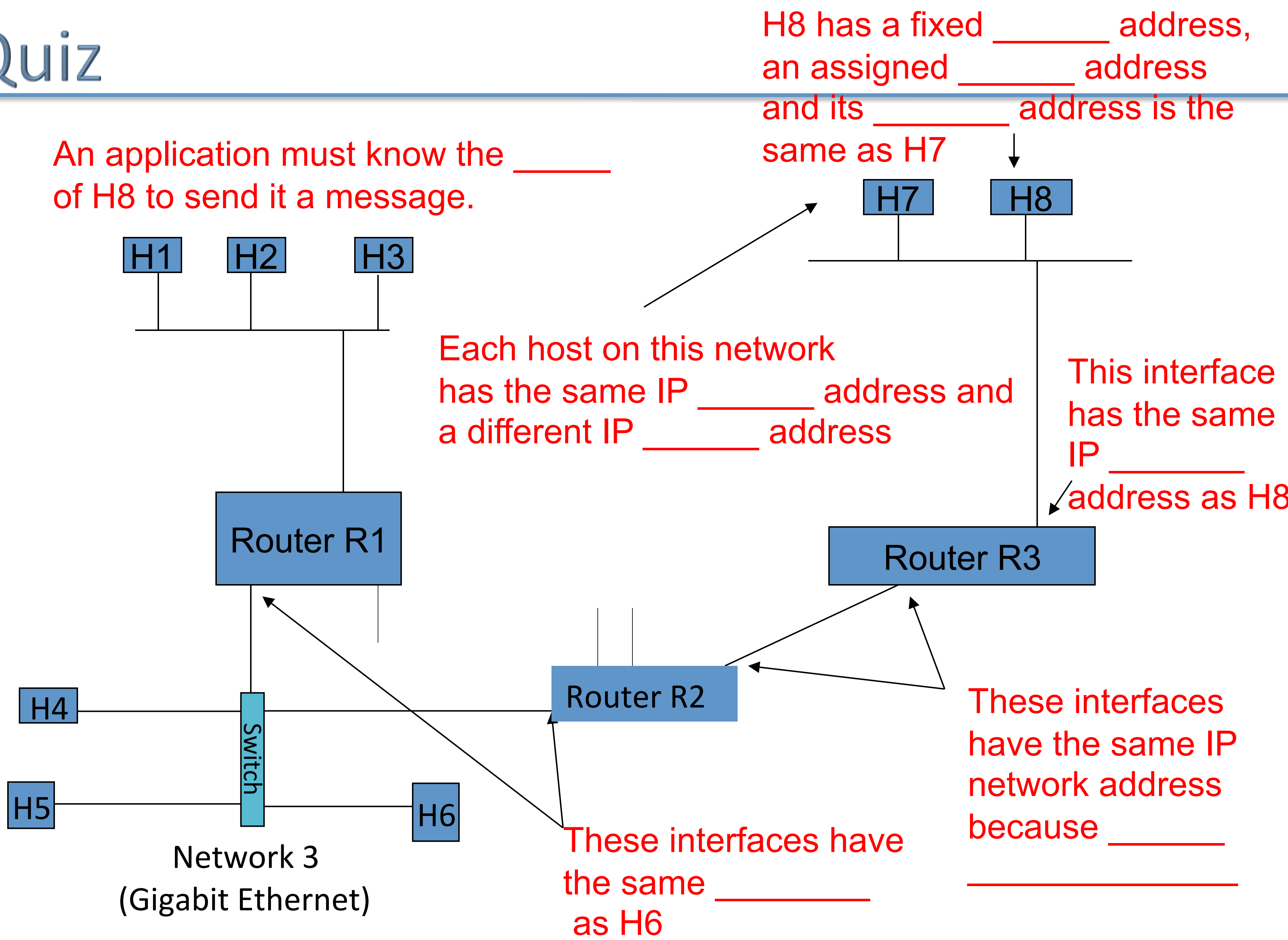
R2 has the packet with the IP address of H8. Does H8 have the same network prefix address as any of my interfaces?
No, so choose the best router - R3.

ARP to H8



R3 has the packet with the IP address of H8. Does H8 have the same network prefix address as any of R3's interfaces?
Yes, so find its Ethernet MAC address via ARP (if necessary) and send the packet.

Quiz

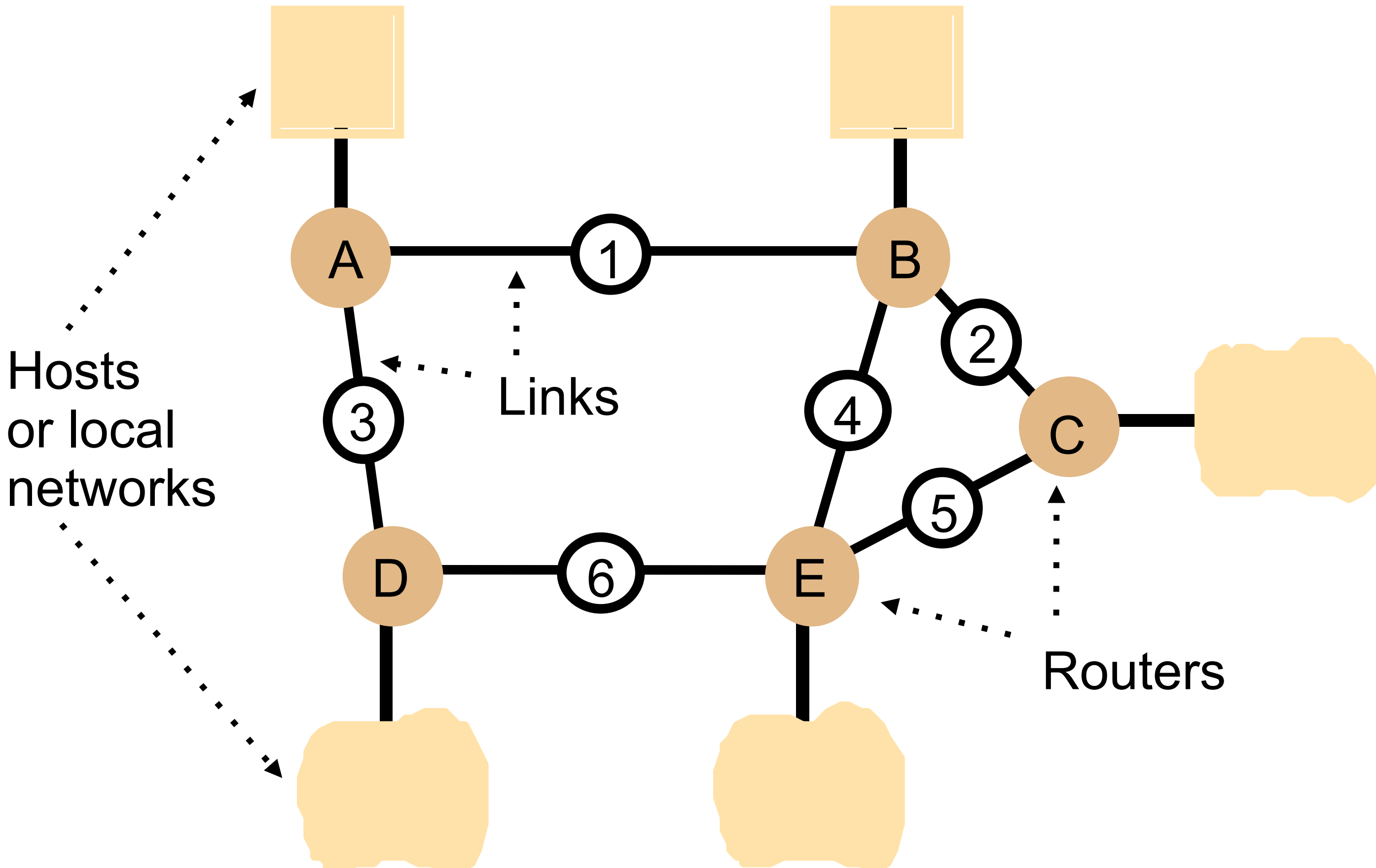

H8 has a fixed _______ address,
an assigned _______ address
and its ________ address is the
same as H7
An application must know the ______
of H8 to send it a message.
H1
H2
H3
H7
H8
Each host on this network
has the same IP _______ address and
a different IP _______ address
This interface
has the same
IP ________
address as H8
Router R1
Router R3
Router R2
H4
H5
H6
Switch
Network 3
(Gigabit Ethernet)
These interfaces have
the same _________
as H6
These interfaces
have the same IP
network address
because _______
____

Routing?

- How did R1 know R2 was the best path to H8?
- My diagrams only showed an obvious path, but but the Internet looks like this:

Source: http://www.cheswick.com/ches/map/movie.mpeg

Routing in a Wide Area Network



- Routers maintain routing tables, indicating the best paths to other networks.

Initial Routing Tables for the Network

Routings from A

To	*Link*	*Cost*
A	local	0
B	1	1
C	-	∞
D	3	1
E	-	∞

Routings from B

To	*Link*	*Cost*
A	1	1
B	local	0
C	2	1
D	-	∞
E	4	1

Routings from C

To	*Link*	*Cost*
A	-	∞
B	2	1
C	local	0
D	-	∞
E	5	1

Routings from D

To	*Link*	*Cost*
A	3	1
B	-	∞
C	-	∞
D	local	0
E	6	1

Routings from E

To	*Link*	*Cost*
A	-	∞
B	4	1
C	5	1
D	6	1
E	local	0

RIP Routing Algorithm

Tl- Table local
Tr- Table remote

***Fault** on n discovered: set cost to ∞ for each destination using that link and execute a send* (Fault: we have not heard from the neighbor on that link for 180 seconds)

Send: Each *t* seconds (typically 30) or when *Tl* changes, send *Tl* on each non-faulty outgoing link.

Receive: Whenever a routing table *Tr* is received on link *n*:

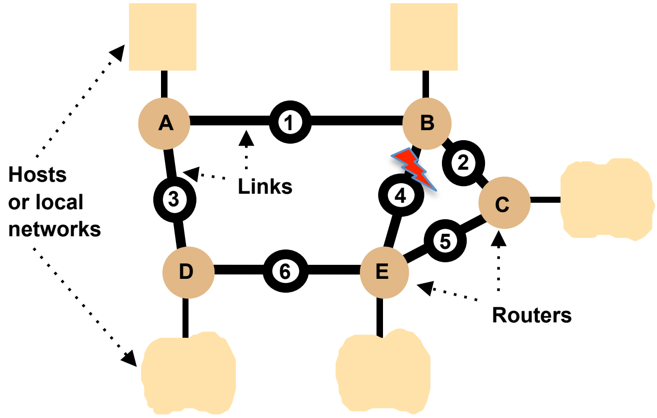
```
for all rows Rr in Tr {
  if (Rr.link ≠ n) {                       // Ignore the route if it was through me, for I know better
    Rr.cost = Rr.cost + 1;                 // This route from me would have higher cost by 1
    Rr.link = n;                           // and I would travel through n
    if (Rr.destination is not in Tl) add Rr to Tl;    // if a new destination, just add to Tl
    else for all rows Rl in Tl {
      if (Rr.destination = Rl.destination and
            (Rr.cost < Rl.cost or Rl.link = n)) Rl = Rr;
                   // Rr.cost < Rl.cost : remote node has better route
                   // Rl.link = n : remote node is more authoritative
    }
  }
}
```

Update A With B

Ta- Local table
Tb- Remote table

Routings from A		
To	*Link*	*Cost*
B	1	1
C	3	3
D	3	1
E	1	2

Routings from B		
To	*Link*	*Cost*
A	1	1
C	2	1
D	-	∞
E	2	2



Receive: Whenever a routing table *Tb* is received on link *n*:
 for all rows *Rb* in *Tb* {
  if (*Rb.link* ≠ *n*) { *// Ignore the route if it was through me, for I know better*
   Rb.cost = *Rb.cost* + 1; *// This route from me would have higher cost by 1*
   Rb.link = *n*; *// and I would travel through n*
   if (*Rb.destination* is not in *Ta*) add *Rb* to *Ta;* *// if a new destination, just add to Ta*
   else for all rows *Ra* in *Ta* {
    if (*Rb.destination* = *Ra.destination* and
      (*Rb.cost* < *Ra.cost* or *Ra.link* = *n*)) *Ra* = *Rb;*
      // Rb.cost < Ra.cost : remote node has better route
      // Ra.link = n : remote node is more authoritative
 }}}

Update A With B

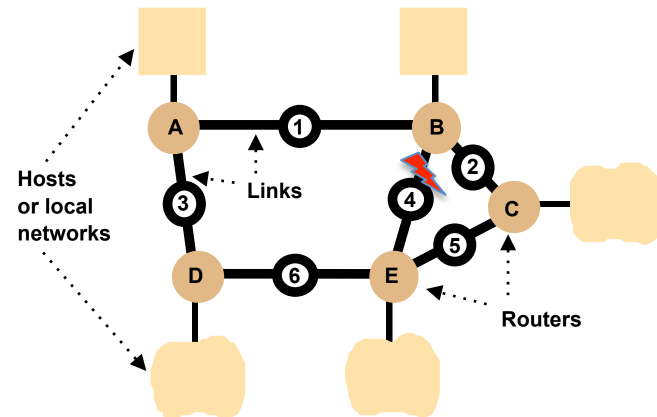
Ta- Local table
Tb- Remote table

Routings from A

To	*Link*	*Cost*
B	1	1
C	1	2
D	3	1
E	1	3

Routings from B

To	*Link*	*Cost*
A	1	1
C	2	1
D	-	∞
E	2	2



Receive: Whenever a routing table *Tb* is received on link *n*:

for all rows *Rb* in *Tb* {

if (*Rb.link* ≠ *n*) { // *Ignore the route if it was through me, for I know better*

Rb.cost = *Rb.cost* + 1; // *This route from me would have higher cost by 1*

Rb.link = *n*; // *and I would travel through n*

if (*Rb.destination* is not in *Ta*) add *Rb* to *Ta;* // *if a new destination, just add to Ta*

else for all rows *Ra* in *Ta* {

if (*Rb.destination* = *Ra.destination* and

(*Rb.cost* < *Ra.cost* or *Ra.link* = *n*)) *Ra* = *Rb;*

// *Rb.cost* < *Ra.cost : remote node has better route*

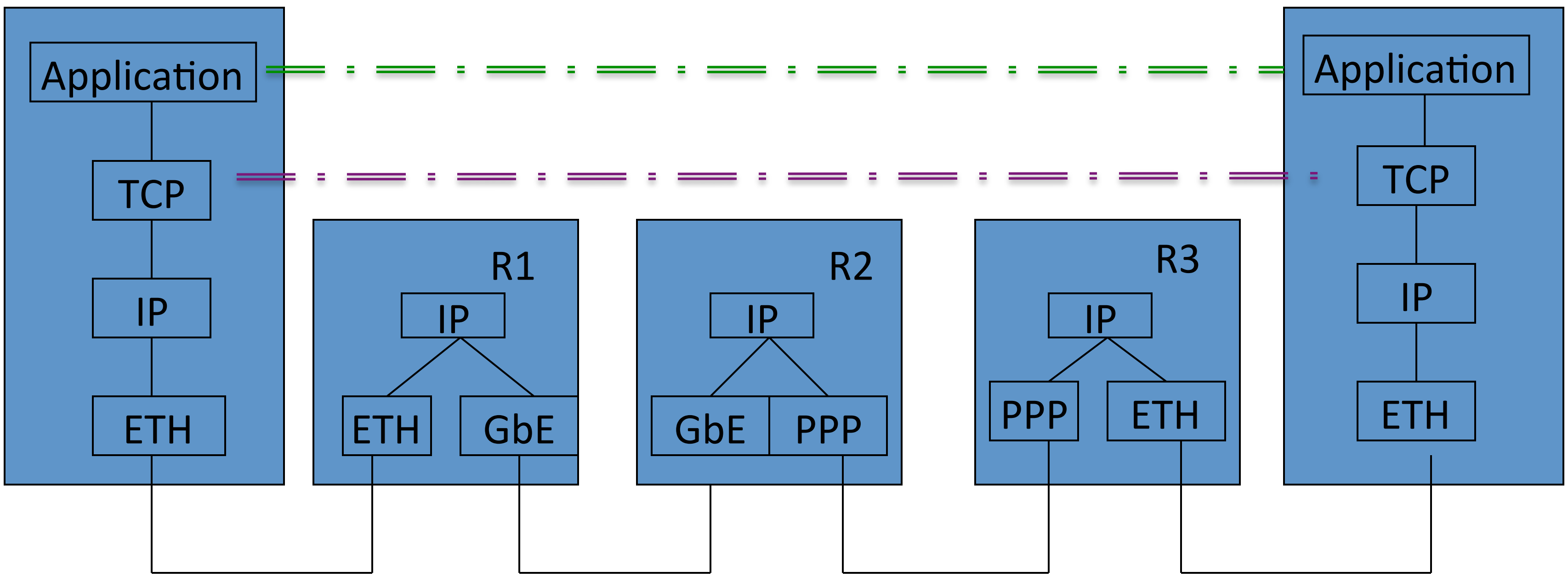
// *Ra.link = n : remote node is more authoritative*

}}}

Types of Routing Algorithms

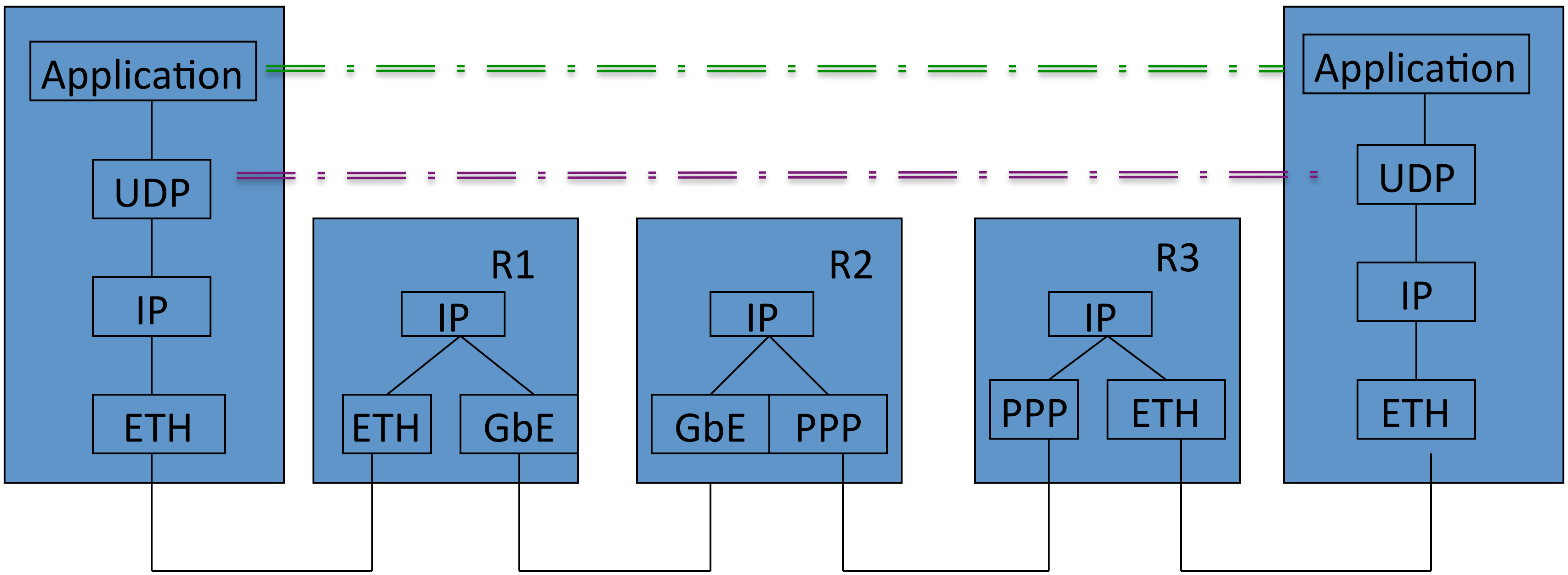
1. Vector-Distance Algorithms
   – RIP is the simplest example
2. Link-State Algorithms
   – Distribution and update of a DB at each node
   – Representing all or a substantial portion of an enterprises' network
   – Each router has a map of the network, with which to calculate shortest path to each other node
   – Collection of shortest paths becomes the routing table

Network Abstractions



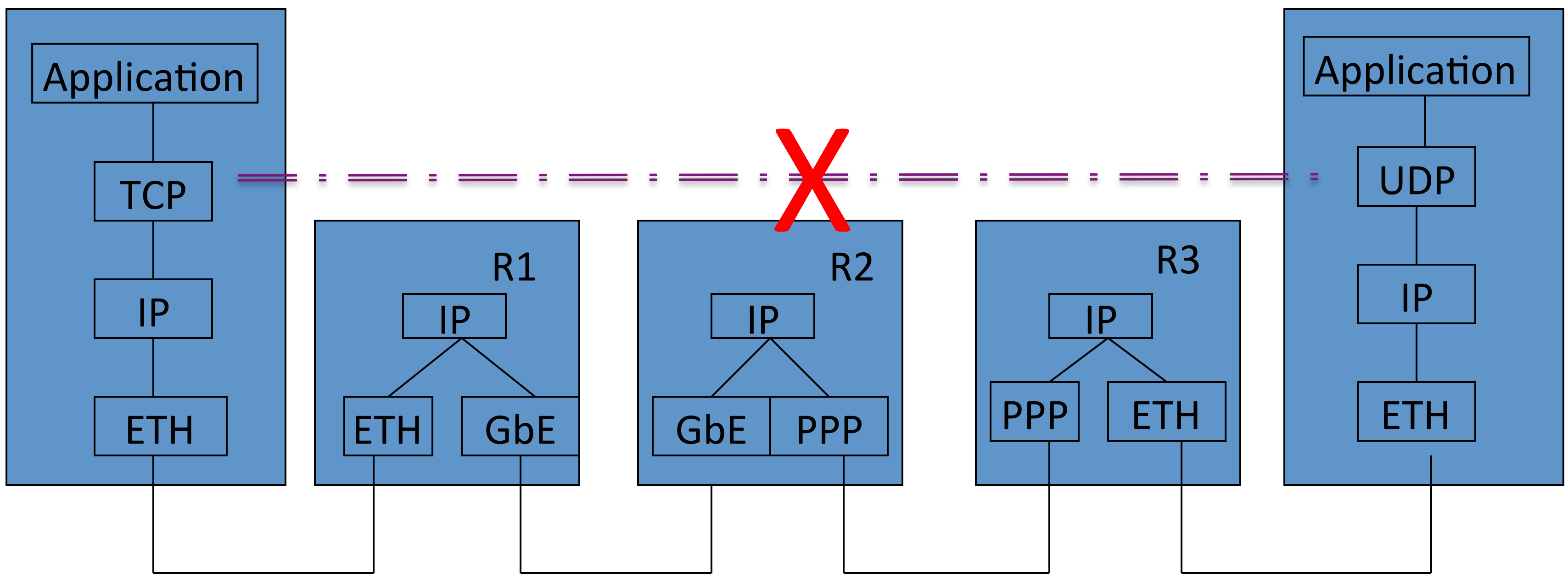
- IP – hosts communicate with routers and other hosts
- TCP & UDP – provides a socket and port abstraction across computers
- Application – application communicates with another application via a sockets connection on a port

Network Abstractions



- Similar with UDP

Network Abstractions



- But they don't mix.

Transport Level Protocols

- Two common transport level protocols
  - UDP
  - TCP
- Both facilitate communication:
  - From a port at one IP address
  - To a port on another IP address
- The communication metaphor between the two ports is called a "socket".
- E.g.
  - A UDP socket
  - from 128.2.25.11 port 5443
  - to 204.100.2.1 port 4568
  - Commonly noted 204.100.2.1:4568
- E.g.
  - A TCP socket
  - Between 128.2.25.22:4567
  - And 204.100.2.2:5432

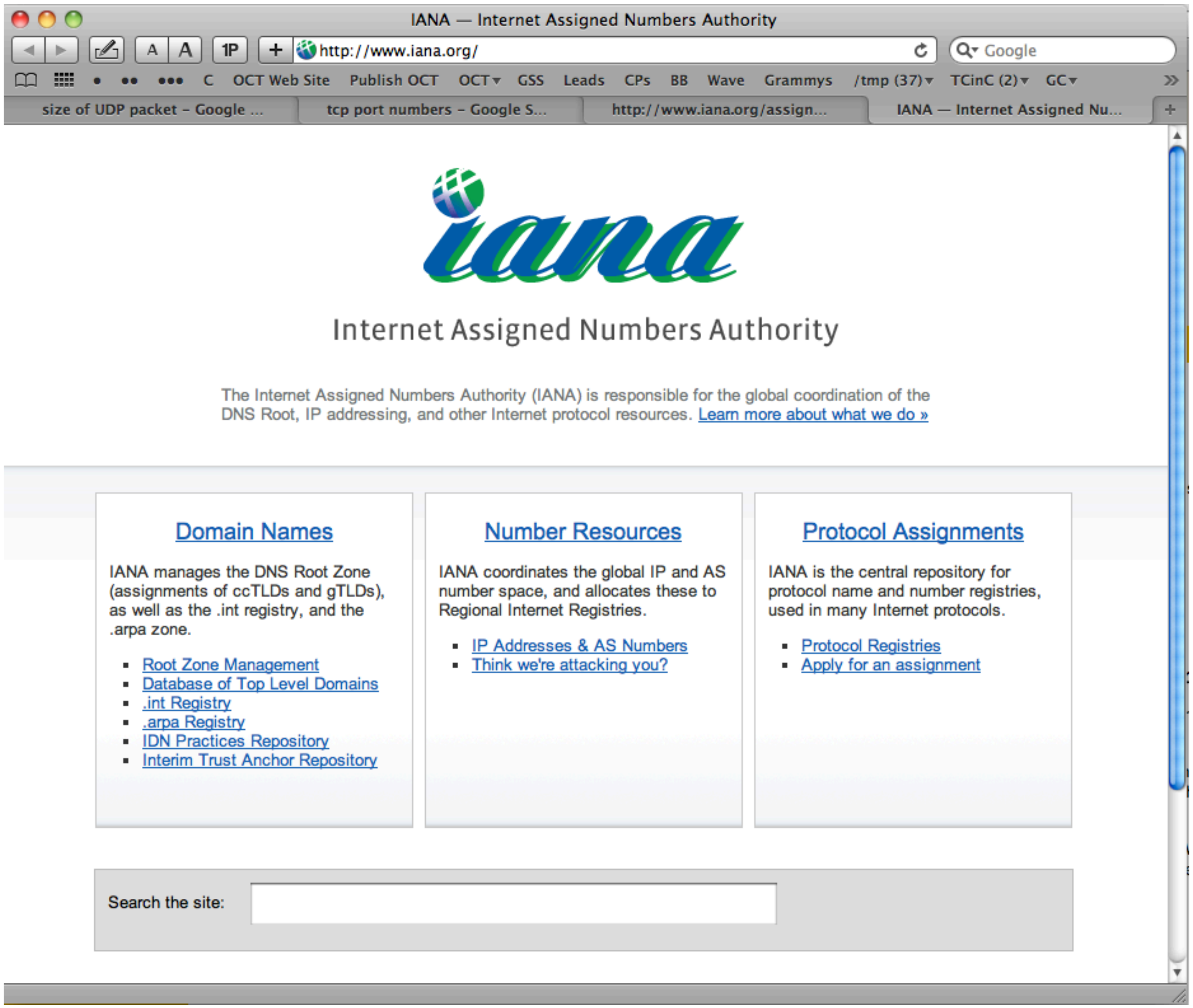
TCP & UDP Comparison

Characteristic	UDP	TCP
General	Simple, high speed, low functionality wrapper that interfaces with IP	Full-featured protocol to reliably communicate data with another application across IP.
Connection	Connectionless, no setup	Connection-oriented; setup prior to transmission
Interface to Application	Discrete message based	Stream based
Reliability and ACK	No reliability guaranteed, best effort delivery without acknowledgement	Reliable, all messages acknowledged
Retransmissions	None. Application must detect if needed	All lost data retransmitted automatically
Flow control	None	Flow control on both ends (sliding window)
Overhead	Low	Low, but not as low as UDP
Speed	Very high	High, but not as high as UDP
Data quantity	Single datagram - up to 65K bytes	Small to very large (gigabytes)
Applicability	Speed matters more than completeness. Small discrete messages. **Multicast or broadcast**	Data must be received reliably, in order.

Ports

- Port is a transport-layer software construct
- Port is a 16 bit integer
- 65,536 different Ports an application can communicate on

Official ports are assigned by IANA



Ports

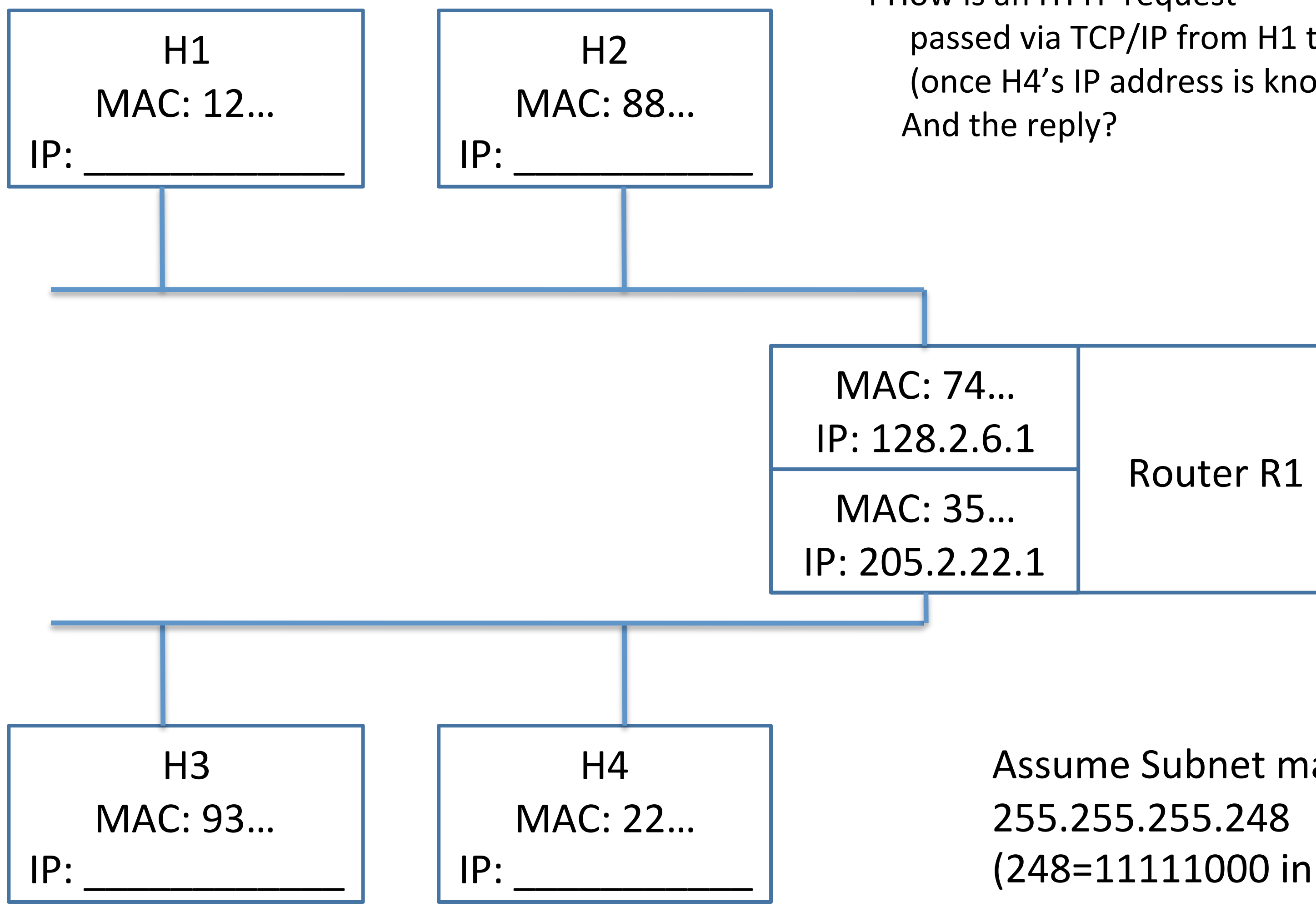
- 0 - 1023 Well Known Ports
  - Should not be used without IANA registration
  - Loosely, system-type-services ports
- 1024 - 49151 Registered Ports
  - Also should not be used without IANA registration
  - Loosely, application-level ports
- 49152 - 65535 Dynamic and/or Private Ports
  - Anything
- Readable list of port assignments:
  - http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

Obvious simple example

- 80 - HTTP
- http://cmu.edu implies http://cmu.edu:80
- You can do http://cmu.edu:12345
  - But you probably won't find http there.

Practice

1) What are valid IP addresses for H1-H4?
2) What are the range of possible IP addresses in each case?
3) How is an HTTP request
   passed via TCP/IP from H1 to H2?
   And the reply?
4 How is an HTTP request
   passed via TCP/IP from H1 to H4?
   (once H4's IP address is known)?
   And the reply?



Assume Subnet masks of 255.255.255.248 (248=11111000 in binary)

Review

- Application layer
  – Application code
  – Alternatively: Application protocols
    • Application code would be above them
- Transport layer
  – TCP and UDP
- Internet layer
  – IP – Internet Protocol
- Data Link Layer
  – MAC - ARP

Some Common Application Protocols

Protocol	Name	Purpose
HTTP	Hypertext Transfer Protocol	Browser to server, web app to web service, etc.
HTTPS	HTTP Secure	Secure HTTP
DHCP	Dynamic Host Configuration Protocol	Server assigns IP address to a host
DNS	Domain Name System	Map domain name to IP address
RTP/RTCP	Real-time Transport Protocol / Real-time Transport Control Protocol	Delivers audio and video streams / statistics and control information about RTP stream
LDAP	Lightweight Directory Access Protocol	Querying and modifying directory services information
POP	Post Office Protocol	Retrieve email from a server
IMAP	Internet Message Access Protocol	Retrieve email from a server
SMTP	Simple Mail Transfer Protocol	Send mail from client to server, and send mail to/from server to server
SSH	Secure Shell	Terminal-type (shell) access to remote computers